Performance Modeling Of The Diffusion Protocol

On Hung (Watson) Wu

A Thesis

in

The Department

of

Electrical and Computer Engineering

Presented in Partial Fulfillment of the Requirements
for the Degree of Master of Engineering at
Concordia University
Montreal, Quebec, Canada

May 1990

ISBN 0-315-64685-3

ABSTRACT

PERFORMANCE MODELING OF THE DIFFUSION PROTOCOL

Diffusion or flood routing has been used as a communication routing technique in defence networks for many years. Its popularity in defence communication networks is due to the following reasons: (1) It provides a very robust environment to the network in terms of connectivity and survivability; (2) With this technique, a network switch requires only a look-up directory of its local subscribers; (3) It always provides the shortest route between the source and the destination switch with respect to the current traffic load on the network. Although diffusion routing has been used for a long time, publications related to the performance of this routing technique are limited.

A network switch has no information about the subscribers located on other switches. Hence, when it receives a call from its subscriber which is not destined to a local subscriber, it will initialize a search message and diffuse it into the entire network. When the destination switch receives this search message, it will send an acknowledgement message back to the source switch through the path just traversed by the search message. Several messages are subsequently exchanged between the source and the destination switch to reserve a virtual circuit for this call. The first search message to arrive at the destination switch determines the route for connection between the source and the

destination switch. The route taken by the first search message varies statistically with the current load of the network, rendering performance analysis of this routing technique very difficult.

The main objective of this thesis is to study the performance of a switching network driven by the diffusion routing technique. The performance analysis is done by a new, innovative iterative technique. Mean value analysis is performed using queueing theory as the mathematical basis. The result of the analysis can be applied to network planning with diffusion routing as the switch routing technique.

Two examples are illustrated. The first one is the mean call setup time, which is the time from the moment the number of the called subscriber is dialed in until the moment the phone of the called subscriber rings. The second example is the mean trunk utilization pattern, which indicates the mean number of channels used in each trunk of the network for call connection.

Simulation results are very promising, with a high degree of correlation to mathematically derived results.

ACKNOWLEDGEMENTS

First, I wish to thank the LORD for giving me the strength, the time, and the inspiration for the completion of this research. My supervisor, Dr Marc A. Comeau, has provided the technical insights that are the driving force behind this research. I have appreciated his kind words in our many discussions.

I would like to thank my colleagues at Canadian Marconi Company for making the office a stimulating research environment. In particular, thank-yous go to Nam Pham for introducing me the wonderful world of Network Simulation and to Jack Kwok and Teddy Beck for proof reading this thesis many times and for their many constructive suggestions.

Last, and certainly not least, I wish to express my deepest gratitude to my family for their encouragement and understanding, and for excusing me from grocery shopping so I would type this thesis.

TO MY PARENTS

Table of Contents

Tables

Figures

CHAPTER I

INTRODUCTION

[illegible] been [illegible]emented to ensure the effi-
c[illegible] and nodal resources in a pa[illegible] Stallings points out in [5], one of the mo[illegible] packet routing technique, i.e. selecting pa[illegible] are to be [illegible] through [illegible] network with the objective [illegible]zing network performance. The performance criteria usually under consideration are the following:

(1) Number of hops - the route through the network with the mi[illegible] number of hops required to pass the packet from the source to the destination switch is sought. (one hop = traversal of one node-to-node link)

(2) Cost - a cost is associated with each link, and the route through the network that accumulates the least cost is sought.

(3) Delay - the route through the network with the shortest delay is sought. The delay is the accumulated sum of queueing and transmission delay.

(4) Throughput - the route through the network with the most efficient use of the current capacity of the network is sought.

With these performance criteria in mind, Kleinrock[3] and Tannenbaum[6] suggest the following basic requirements for a good routing technique:

(1) It should ensure rapid and reliable delivery of messages.

(2) It should adapt to varying source - destination traffic loads.

(3) It should adapt to changes in network topology resulting from nodal and link failure.

(4) It should route packets away from temporarily congested nodes within the network.

(5) It should determine the connectivity of the network.

(6) It should allow easy and "automatic" insertion and deletion of nodes.

(7) It should not provide a trade-off between fairness and optimality. In other words, it should not favor the exchange of packets between nearby stations and discourage the exchange between distant stations.

In an idealized situation where all parameters of the network are assumed to be known and not changing, it is possible to determine a routing strategy which satisfies the above routing requirements, and at the same time optimizes the network performance. However, in a practical application, many parameters of the network are not known and are changing. As a result, the performance of any practical routing technique is very difficult to predict and the network planner always faces a tough problem in trying to optimize network performance.

H. Rudin has placed the routing problem in a more formalized framework in Rudin[8]. The routing problem is a problem distributed in space and time. Consider a network consisting of N nodes arbitrarily interconnected. At any time t, there is a certain amount of information, $w_{ij}(t)$, which is the information currently at node i with final destination j. $w_{ij}(t)$ can be the number of packets required to be delivered from switch i to j at time t. The elements $w_{ij}(t)$ form a N x N matrix W(t) which describes the "work" which the network has to do at time t. Note that W(t) can change very rapidly in time. Similarly, there is an array R(t) which describes the resources available in the network to do the work W(t). The elements $r_{ij}(t)$ can be the information carrying capacity of the communication link directly connecting node i to node j or the delay associated with each link. Then the routing problem consists of how to best allocate the resources R to accomplish the work W. "Best" is taken here to mean the degree of satisfaction of the performance criteria described earlier. What makes the design of a routing technique a difficult problem is that the knowledge of W(t) and R(t) exist in distributed fashion. In other words, the individual work loads $w_{ij}(t)$ are known accurately at time t only at the switch i. This information can be passed on to another node but by the time it arrives at this second node it is most likely out of date. Hence, in all networks of nontrivial interest, any global characteri 'tion of the network load W(t) can be based only on past as opposed to current information.

A number of techniques have been proposed to solve the routing problem. There are a number of ways to categorize these routing techniques, such as virtual circuit versus datagram implementation, static versus dynamic response to changing conditions, and centralized versus distributed control. Stallings[5] and Tannenbaum[6] have given a lengthy description on these different techniques. Kleinrock[2], Kleinrock[3] and Hayes[4] have provided the approximated performance on some of these techniques. The aim of this thesis is not to survey these techniques. Instead, a study of virtual circuit with dynamic response diffusion routing will be studied. The details of this routing technique will be elaborated on in the following section.

Diffusion, or flooding, has been used as a communication routing technique in defence networks for many years. Its popularity in defence communication networks is probably due to the following reasons:

(1) The diffusion routing provides a very robust environment for the network; even if a few switches and trunks are removed from the network, the network can still operate.

(2) Using this technique, a network switch does not require a look-up table for routing the incoming messages. Therefore, even if a switch is captured by the enemy, the security of the network will still be preserved.

(3) This routing technique always provides the shortest route between the source and the destination switch with respect to the current traffic load of the network.

Although the diffusion routing technique has been used for a long time, publications related to the performance of this technique are limited because it is used primarily in Defence, making much of the information classified. Also, since messages are duplicated inside the network during the diffusion routing, the product-form solution proposed by Jackson's theory in Jackson[13] does not apply. Finally, due to the complexity of the technique, many results of traditional queueing theory do not apply. As a result, attempts to study this technique are presently only done by computer simulation. However, as mentioned in Gelenbe[7], a communication network is composed of many varying parameters, and each of them may vary at different rates. It will be necessary to simulate the network at the time scale which corresponds to the rapidly varying parts in order to preserve the desired accuracy. Yet, the total simulation time will have to be large compared to the slowly varying parts in order for the simulation to reach steady state.

Furthermore, the probabilistic or statistical tools available at present do not allow us to estimate accurately the confidence intervals of the simulation results. There are two major difficulties. The first one is the statistical undesirable effect produced during the transient period of the simulation. It is customary to initialize a simulation with the simulation system in an empty or idle state. It is especially true for simulating a communication system because its operating state is usually unknown before hand. It takes some time for the model to "warm-up" and the values of the output variables collected during this warm period may not be representative of the steady-state behavior, which is the target of the measure of the confidence interval. Therefore, in order to estimate the confidence interval accurately, the length of the transient period has to be estimated accurately and the output collecting process has to bypass this period. However, MacDougall[15] and Lavenberg[16] have demonstrated that the current mathematical tools have not provided any reliable method to estimate the length of the transient period. The only result available about transient behavior is that of the M/M/1 queue, which has recently been studied by Cantrell[17] and will be used later to estimate the simulation outputs.

There are many methods designed to circumvent this "warm-up" problem, such as (1) replication, (2) batch means, (3) regeneration, (4) autoregression, (5) spectral analysis, (6) standardized time series. MacDougall[15], Lavenberg[16], and Law[18] give a detailed account of these methods. The principle behind these methods is to try to be generous in simulation run time (computing time). Since the diffusion routing is very complex and may take about 75 hours of computing time from a dedicated VAX station just to simulate 7500 seconds of moderate traffic, the limited computing resources make these methods inapplicable to the current study.

The second difficulty is the input correlation inherent in the multi-queue environment. The method of confidence interval estimation is based on the concept that the samples measured should be independent of each other. However, in the multi-queue environment such as in diffusion routing, the simulated output variables are correlated and the requisite assumption of independence does not hold.

1.2 OUTLINE OF THE THESIS

The main objective of this thesis is to study the performance of a switching network driven by the diffusion routing technique. The performance analysis employs queueing theory, and it is limited to the mean value analysis. The discussion is divided into four parts which:

(1) Describe the operation of the diffusion routing and the concept of switching networks.

(2) Construct a mathematical model of diffusion routing in a switching network.

(3) Analyze the performance of a switching network under diffusion routing.

(4) Present the conclusion of this analysis and elaborate on the outstanding problems.

CHAPTER II

DIFFUSION ROUTING AND SWITCHING NETWORKS

2.1 PRELIMINARIES

The switching network considered here is composed of a number of switches and trunks. The network switch studied resembles a commerical switch in many aspects, except that a network switch does not have the look-up table to direct incoming calls. Inside the network, telephones may connect to a multiplexer which, in turn, connects to a switch, or they may connect directly to a switch. When a call is generated by a telephone, the called number will be directed to the switch's processor. If the called subscriber (telephone) is also connected to this switch (local call), the switch's processor will try to establish a connection between these two subscribers. However, if the call is not a local call, the called subscriber will be located in another switch of the network. Diffusion routing will then be used by the switch to locate and try to set up a connection with the called subscriber.

Diffusion routing is designed such that switches inside a network do not require a look up table for routing the incoming messages. When a switch receives a call from its subscribers, it will initialize a search message and diffuse it into the entire network. When the destination switch receives this search message, it will send an acknowledgement message back to the source switch through the path just traversed by the search message. Several messages are subsequently exchanged between the source and the destination switch to reserve a virtual circuit for this call.

The diffusion search considered here is called, ***selective*** diffusion search, because a switch does not always diffuse a received search message to all of its connected trunks. The selective diffusion process is carried out according to the following rules:

(1) If the switch is the originator of this search message, it will not continue to diffuse it.

(2) If the switch is the destination of this search message, it will not continue to diffuse it.

(3) If the switch is neither the destination nor the originator of this search message, and has received the same search message before, it will not continue to diffuse it.

(4) If the switch is neither the destination nor the originator of this search message and this is the first time it receives this search message. It will modify this message and transfer it to all of its output queues except the one to the node from which this search message arrived. This is because all the trunk lines in the network are assumed to be duplex, then allowing the message transfer to the node from which the search message arrived will mean flooding the search message toward the originator switch.

Thus, each call arriving at a switch causes a diffusion of the search message to the whole network. Therefore, the process of diffusion search places a significant overhead on the network traffic.

Each switch connects to the other switches in the network through trunk lines. All the trunk lines are assumed to be operated in the duplex fashion. The network traffic can be either data or voice. Each switch has a number of input/output ports used for trunk connections. There is an output queue associated with each input/output port containing messages waiting to be transmitted through the associated trunk. Its transmission time depends on the Bit Error Rate (BER) associated with the trunk and its transmission rate. When a message enters a switch through one of the input/output ports (duplex trunks), it will be queued up in a central queue waiting to be processed by the switch's processor. After being served by the processor, the message may be sent to the output queue(s), or to one of the local subscribers of the switch.

The performance criteria stressed by the diffusion routing technique is to find the path with the shortest delay (queueing and transmission delay) from the source switch to the destination switch. Using the requirements of good routing techniques given in the beginning, the following will further elaborate on how this routing technique satisfies these requirements and compares its efficiency with the other dynamic routing techniques.

(1) When compared with the other dynamic routing techniques with built-in routing tables, this technique has a slower delivery of call set-up messages. The main reason is that the switch lacks a look-up table for routing, hence it requires more messages to set up a call. Also, every call generates a flood search to the entire network, placing a heavy overhead in terms of delay on the network.

(2) The traffic adaptation of diffusion routing is done on a call basis. The flood search initialized in the beginning of routing can automatically set up a virtual circuit away from the congested region.

(3) Adding and deleting nodes from the network is relatively easy using diffusion routing. Only the intended connected nodes are affected when adding or deleting a node.

(4) The network topology and connectivity are adapted to by the flood search. If a trunk line has failed, the search message will not pass this trunk during diffusion search. As a result, the failed trunk will not be used for setting up a call.

Before entering into the analysis, it is important to list the mathematical difficulties confronted by the queueing theory in analyzing this routing technique. With the understanding of these difficulties, the reader can better appreciate the assumptions made later in the analysis section. The following lists the difficulties:

(1) The routing technique responds continuously to the changes in the network load. The quasistatic environment mentioned in Hayes[4] that is essential to stochastic analysis does not apply in this context.

(2) Since messages are duplicated inside a network during the diffusion search, one of the fundamental assumptions, conservation of incoming and outgoing flows, based on Jackson theory in Jackson[13] does not hold with this technique.

(3) Since every call generates a flood search to the whole network, there will be tight interference between all the output queues of the network.

(4) The trunk lines are restricted to a number of available channels for circuit switching. These channels can be used for either data or voice traffic. Call blocking is possible and will place a heavy burden on the mathematical analysis.

(5) If a search message entering into an output queue eventually leads to a successful call set-up, a "bursty" arrival of call set-up messages is expected to this queue in a short time. As a result, the arrival rate will not be a Poisson process and many queueing results can not be applied.

(6) There are a finite number of constant length messages circulating inside the network. These messages can be the call set-up messages or the search message. With this kind of arrival process, the queueing analysis belongs to multi-class queueing network theory. The analysis of this type of queue is very difficult and the output will not be Poisson.

2.2 THE MATHEMATICAL MODEL OF DIFFUSION ROUTING

2.2.1 GENERAL DESCRIPTION

The objective of this section is to derive a set of equations describing a switching network operating under the diffusion protocol. The physical implications of employing diffusion in switching networks have already been discussed in the previous section. The set of equations derived in this section is used to solve for the mean traffic intensity factor associated with each queue inside a switching network. The derivation process requires first building the mathematical model of a switching network, then formulating a set of equations to describe the network model under diffusion search (the initialization process of diffusion routing). Based on the insights generated in the first part, approximations are proposed to modify the set of equations for diffusion search such that it becomes applicable to general diffusion routing. Finally, two characteristics of the network model are calculated using the results in the previous part. These two characteristics are the mean call set-up time and the mean trunk utilization pattern.

2.2.2 THE MATHEMATICAL MODEL OF A SWITCHING NETWORK

The physical implementation of the switching network and the diffusion routing technique have already been discussed in the previous sections. The corresponding mathematical model will now be developed.

When a switching network is composed of N switches, its state, S can be denoted as :

$$S_N = (s_1, s_2, \ldots, s_N);$$

where s_i represents the configuration of switch i in the network. Each trunk is assumed to be operated in the duplex fashion and is modelled as a server in terms of queueing theory. An unlimited number of channels is assumed for each trunk in order to simplify the analysis. This assumption will be re-considered in the later sections. The processing time of the switch's processor is negligible compared to the transmission time of the trunk server. Therefore, the analysis of the central queue is ignored.

Based on the switch model described in the previous section, without considering the central queue, the configuration of switch i, s_i, can be denoted as:

$$s_i = (q_{i1}, q_{i2}, \ldots, q_{ip(i)});$$

where p(i) is the number of the input/output ports possessed by switch i and q_{ij} denotes the output queue j of switch i. The following three terms are defined for the queue q_{ij} :

λ_{ij} is the mean arrival rate to this queue;

$1/\mu_{ij}$ is the mean service time of the trunk server associated with this queue;

ρ_{ij} is the traffic intensity factor associated with this queue;

The derivation process of the mathematical model is divided into 2 parts:

(1) build the mathematical model of the diffusion search.

(2) extend the model of diffusion search to study the diffusion routing technique.

Two key assumptions are applied to the mathematical model such that the analysis can be mathematically tractable. The two assumptions are the following:

(a) Kleinrock's independence assumption

Each time a message joins the output queue of a switch, its length is determined afresh from the probability density function

$$b(x) = \mu_{ij} \exp(-\mu_{ij} x), \; x >= 0;$$

where $1/\mu_{ij}$ is the mean message transmission time. If the external arrivals to the network is assumed to be under the Poisson distribution, then the M/M/1 queue analysis can be applied to each output queue of the network.

(b) Nonpassing condition

The nonpassing condition described in Lam[9], Wong[10] and Wong[11] is assumed to be applicable to the network. π_k is defined as an arbitrary path in a network. The nonpassing condition means that for each pair of channels i, j in π_k, a packet arriving after any tagged packet k at channel i will never overtake this tagged packet at channel j. In other words, for two queues q1 and q2 connected by two different chains i and j, if a packet l arrived after any tagged packet k in queue q1, and packet l traverses through chain i and tagged packet k traverses through chain j, then when these two packets arrive to queue q2, the nonpassing condition states that packet l will never arrive before the arrival of the tagged packet k. With this condition, the following Laplace transform of the distribution of path delay results as shown in Lam[9]

$$T_k^*(\lambda) = \prod_{ij \, \varepsilon \, \pi_k} [(\mu_{ij}(1 - \rho_{ij})) \; / \; (\lambda + \mu_{ij}(1 - \rho_{ij}))]$$

where μ_{ij} and ρ_{ij} are the mean service time and the traffic intensity factor of the output queue j of node i. T_k^* is the Laplace transform of the path delay distribution of path π_k, which is composed of a set of output queues.

For those chains where the nonpassing condition is not satisfied, simulation experiments described in Wong[10] have shown that the end-to-end delay formula derived based on this condition still gives accurate approximations.

2.2.3 SWITCHING NETWORKS UNDER DIFFUSION SEARCH

The diffusion search initialized by any switch contributes traffic to all the switches in the network. Therefore, a tightly-coupled relationship of traffic exists among all the switches in the network.

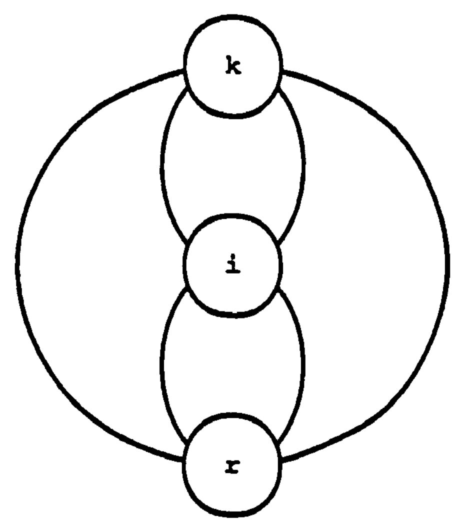

Figure 1. The network graph illustrating the connection between the source switch k and the destination switch r with i as one of the intermediate switches.

The analysis begins by studying the effect on the volume of traffic entering an arbitrary switch i induced by the diffusion search generated by switch k to locate switch r. Then the same traffic analysis applies to each switch in the network. As a result, the tight traffic relationship among the switches can be quantified into tractable mathematical form. In the following analysis, the message length of the search message is assumed to be under the negative exponential distribution with a mean equal to $1/\mu$ and each link is assumed to behave as an M/M/1 queueing system.

Switch i is one of the intermediate switches in some of the paths connecting switch k to switch r. As we focus on switch i, the contributions of the search message generated by switch k to different output queues of switch i are determined by the port to which the search message arrived first, because all identical search messages, arriving after the first one, are ignored and have no traffic effect on the output queues. The relationship between the port to which the search message arrived first and the traffic contribution by the search message are illustrated in Figure 2. (switch i is assumed to have 3 active ports)

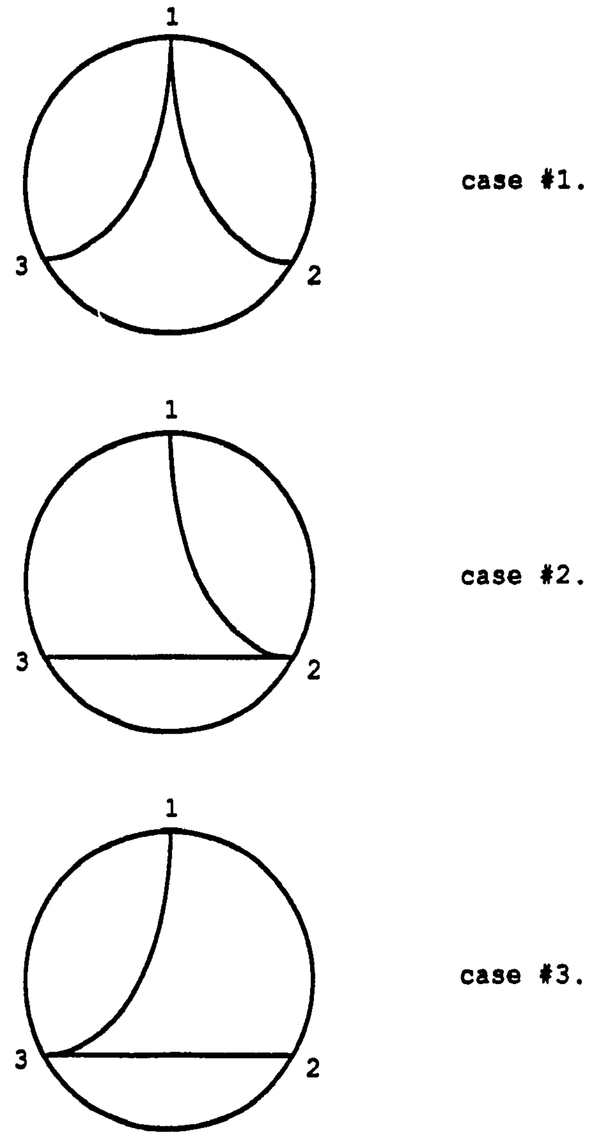

Figure 2. The diffusion of a search message inside a switch.

In case 1, the search message arrives to port 1 first. This search message is duplicated and put into the output queues associated with ports 2 and 3. In case 2, the search message arrives to port 2 first. This search message is duplicated and put into the output queues associated with ports 1 and 3. In case 3, the search message arrives to port 3 first. This search message is duplicated and put into the output queues associated with ports 1 and 2.

Two observations result from Figure 2:

(a) The probability of the search message entering the output queue associated with port i is equal to the probability that the first search message will not arrive at the switch through port i.

(b) The probability of the search message entering the output queue associated with port i is equal to the sum of the probabilities that the first search message will enter switch i through each of its ports, except i.

The above two observations will be formalized mathematically later. The port at which the search message arrived first is determined by the path selected by the search message travelling from switch k to switch i. The selected path is the shortest out of all the possible ones. A path in the network is conceptualized as a sequence of input/output queues, such as the path illustrated in Figure 3, which

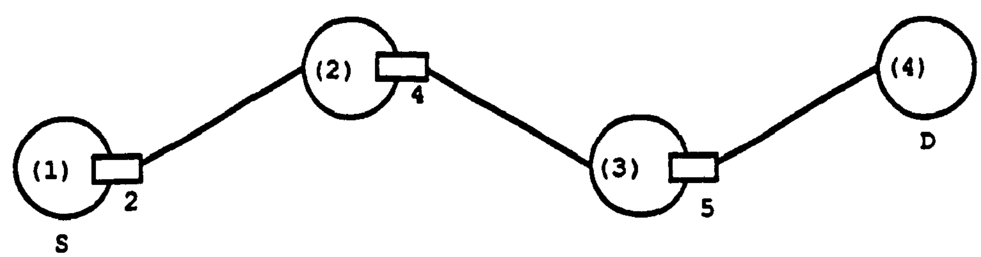

Figure 3. A path in the network represented by a sequence of input/output queues.

will be specified as (q_{12}, q_{24}, q_{35}).

The following terms are introduced to formulate the diffusion search into mathematical form:

λ_k the mean external arrival rate to switch k;

P_{kr} the probability of switch k calling switch r;

λ_{ij}^{kr} the mean arrival rate to the output queue j of switch i due to the search messages originated by switch k in search of switch r;

$Path_{ki}$ the set of paths connecting switch k to switch i.

$Path_{kr}^{[ij]}$ the set of paths connecting switch k to switch r which do not contain the output queue j of switch i.

$Path_{kr}^{\{ij\}}$ the set of paths connecting switch k to switch r which contain the output queue j of switch i.

$Path_{kr}^{[i]}$ the set of paths connecting switch k to switch r not containing switch i as an intermediate switch;

$Path^{i}_{kr}$ the set of paths connecting switch k to switch r containing witch i as an intermediate switch;

It is obvious that $Path_{kr} = Path^{i}_{kr} \cup Path^{[i]}_{kr}$.

$Path^{im}_{kr}$ the subset of $Path^{i}_{kr}$ containing those paths that terminate at port m of switch r;

$Path^{[i]m}_{kr}$ the subset of $Path^{[i]}_{kr}$ containing those paths that terminate at port m of switch r;

$Prob^{ic}_{kr}$ the probability that path c is the shortest one out of the set $Path^{i}_{kr}$;

$Prob^{[i]c}_{kr}$ the probability that path c is the shortest one out of the set $Path^{[i]}_{kr}$;

The following mathematical relationships hold with respect to the above probability and path definitions.

$$\bigcup_{m=1}^{p(i)} \text{Path}_{kr}^{[i]m} = \text{Path}_{kr}^{[i]}; \quad \bigcup_{m=1}^{p(i)} \text{Path}_{kr}^{im} = \text{Path}_{kr}^{i} ;$$

and the intersection between $\text{Path}_{kr}^{[i]m}$ and $\text{Path}_{kr}^{[i]n}$ is an empty set when m is not equal to n.

$$\sum_{c \,\varepsilon\, \text{Path}_{kr}^{[i]}} \text{Prob}_{kr}^{[i]c} = 1; \quad \sum_{c \,\varepsilon\, \text{Path}_{kr}^{i}} \text{Prob}_{kr}^{ic} = 1;$$

With the above definitions, the two observations made earlier relating the port to which the search message arrived first to the traffic contribution to the other queues at that switch takes the following mathematical form. Focusing on output queue j of switch i for messages originating at switch k in search of switch r, we consider all paths from k to i which exclude switch r as an intermediate switch.

$$\sum_{\substack{c = 1; \\ c \neq j}}^{p(i)} \sum_{t \,\varepsilon\, \text{Path}_{ki}^{[r]c}} \text{Prob}_{ki}^{[r]t} = 1 - \sum_{t \,\varepsilon\, \text{Path}_{ki}^{[r]j}} \text{Prob}_{ki}^{[r]t}$$

The left value is equal to the sum of the probabilities that the first search message enters into switch i through each of its ports, except j, while the right value is equal to the probability that the first search message does not arrive at the switch through port j. The value of $Prob_{ki}^{[r]t}$ is equal to $\prod_{z \in Path_{ki}^{[r]};\, z \neq t} Pr(\pi_z >= \pi_t)$, where π_z means path z. The probabilty that the path π_t is shorter than or equal to path π_z ($Pr(\pi_z >= \pi_t)$) is defined in APPENDIX A and its resultant form is shown in the following:

$$= \sum_{k \in \pi_z} \sum_{l \in \pi_t} [(\sigma_k^{\pi_z} . \sigma_l^{\pi_t})(\mu - \lambda_l)] / [(\mu - \lambda_l)(\mu - \lambda_k)(2\mu - \lambda_l - \lambda_k)]$$

with $\sigma_i^{\pi_r} = [\prod_{k \in \pi_r} (\mu - \lambda_k)] / [\prod_{k \in \pi_r;\, k \neq i} (\lambda_i - \lambda_k)]$

The following will show that the sum of probabilities $Prob_{ki}^{[r]t}$ is equal to one.

$$\sum_{t \in Path_{ki}^{[r]}} Prob_{ki}^{[r]t} = \sum_{t \in Path_{ki}^{[r]}} \prod_{z \in Path_{ki}^{[r]},\, z \neq t} Pr(\pi_z >= \pi_t) = 1;$$

Let's assume there are 2 paths, π_z and π_t, connecting switch k to i, and λ_j is the mean arrival rate to switch j. Then

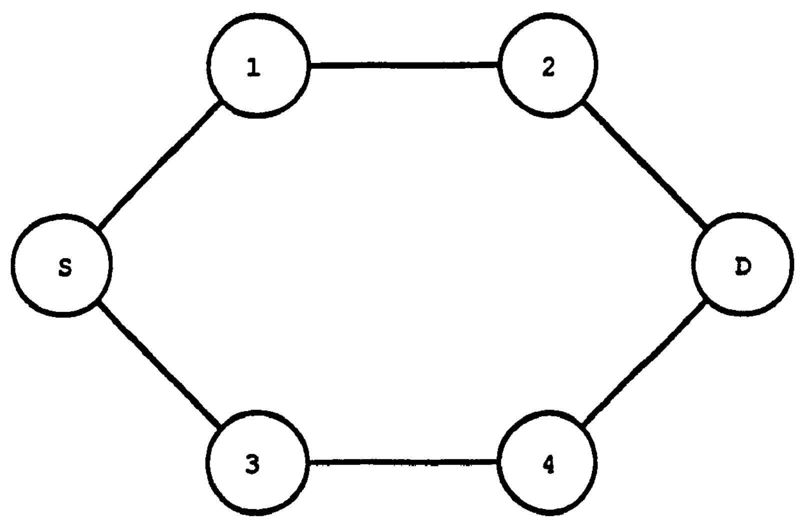

Figure 4. A network example used to illustrate the path delay computation between switches.

$$\sum_{t \, \varepsilon \, Path_{ki}^{[r]}} Prob_{ki}^{[r]t} =$$

$$\sum_{k \, \varepsilon \, \pi_z} \sum_{l \, \varepsilon \, \pi_t} [(\sigma_k^{\pi_z} . \sigma_l^{\pi_t})(\mu - \lambda_l)] \, / \, [(\mu - \lambda_l)(\mu - \lambda_k)(2\mu - \lambda_l - \lambda_k)] +$$

$$\sum_{k \, \varepsilon \, \pi_z} \sum_{l \, \varepsilon \, \pi_t} [(\sigma_k^{\pi_z} . \sigma_l^{\pi_t})(\mu - \lambda_l)] \, / \, [(\mu - \lambda_l)(\mu - \lambda_k)(2\mu - \lambda_l - \lambda_k)]$$

After simple algebraic manipulation, the following form results:

$$= \sum_{k \, \varepsilon \, \pi_z} \sum_{l \, \varepsilon \, \pi_t} [\sigma_k^{\pi_z} . \sigma_l^{\pi_t}] \, / \, [(\mu - \lambda_l)(\mu - \lambda_k)]$$

Substituting the parameters of the two paths,

$$\sigma_1^{\pi_z} = (\mu - \lambda_1)(\mu - \lambda_2) \, / \, (\lambda_1 - \lambda_2)$$

$$\sigma_2^{\pi_z} = (\mu - \lambda_1)(\mu - \lambda_2) \, / \, (\lambda_2 - \lambda_1)$$

$$\sigma_3^{\pi_t} = (\mu - \lambda_3)(\mu - \lambda_4) \;/\; (\lambda_3 - \lambda_4)$$

$$\sigma_4^{\pi_t} = (\mu - \lambda_3)(\mu - \lambda_4) \;/\; (\lambda_4 - \lambda_3)$$

$$\sum_{k \,\varepsilon\, \pi_z} \; \sum_{l \,\varepsilon\, \pi_t} [\sigma_k^{\pi_z} . \sigma_l^{\pi_t}] \;/\; [(\mu - \lambda_l)(\mu - \lambda_k)] = 1$$

If the mean arrival rate of the output queues of all the switches are known except for switch i, then the mean arrival rate to queue j of switch i due to the traffic from switch k to switch r is:

$$\lambda_{ij}^{kr} = \sum_{\substack{c = 1 \\ c \neq j}}^{p(i)} \; \sum_{t \,\varepsilon\, \mathrm{Path}_{ki}^{[r]c}} [(\mathrm{Prob}_{ki}^{[r]t})(P_{kr})(\lambda_k)]$$

The mean arrival rate to output queue j of switch i is then obtained by summing the traffic contribution by all the other switches in the network plus its own contribution.

$$\lambda_{ij} = \sum_{\substack{r = 1; \\ r \neq i}}^{N} \; \sum_{\substack{k = 1; \\ k \neq i}}^{N} \lambda_{ij}^{kr} + \lambda_i$$

The traffic intensity factor, ρ_{ij}, is equal to

$$\rho_{ij} = \lambda_{ij} / \mu$$

The same procedure used to derive λ_{ij} is applied to every output queue in the network. As a result, a set of $\sum_{i=1}^{N} p(i)$ nonlinear equations is generated. The mean arrival rate to each queue in the network can be found by solving the following set of equations iteratively.

$$\mathrm{Prob}_{ki}^{[r]t} = \prod_{\substack{z \,\varepsilon\\ \mathrm{Path}_{ki}^{[r]};\, z \neq t}} \Pr(\pi_z >= \pi_t)$$

$$\mathrm{Prob}_{ki}^{[r]t} = \prod_{\substack{z \,\varepsilon\\ \mathrm{Path}_{ki}^{[r]};\, z \neq t}} \sum_{\substack{k \,\varepsilon\\ \pi_z}} \sum_{\substack{l \,\varepsilon\\ \pi_t}} \{ [(\sigma_k^{\pi_z} \cdot \sigma_l^{\pi_t})(\mu - \lambda_l)] / [(\mu - \lambda_l)(\mu - \lambda_k)(2\mu - \lambda_l - \lambda_k)] \}$$

$$\lambda_{ij}^{kr} = \sum_{\substack{c = 1\\ c \neq j}}^{p(i)} \sum_{\substack{t \,\varepsilon\\ \mathrm{Path}_{ki}^{[r]c}}} [(\mathrm{Prob}_{ki}^{[r]t})(P_{kr})(\lambda_k)]$$

$$\lambda_{ij} = \sum_{\substack{r = 1; \\ r \neq i}}^{N} \sum_{\substack{k = 1; \\ k \neq i}}^{N} \lambda_{ij}^{kr} + \lambda_i$$

with $1 <= i <= N$;

$1 <= j <= p(i)$;

Before iterating the above system of nonlinear equations, a set of initial values (the arrival rate to each output queue of the network) has to be determined. This set of initial values determines the convergence point (the steady state values) if the system of equations has more than one solution, and also controls speed of convergence (the number of iterations) before a stable point is reached.

The set of initial values is generated by replacing the equation for computing the probability Prob ($\pi_g > \pi_z$) with a simpler measure. Instead of using the statistical delay function illustrated in Lam[9] as the basis, the concept of hop count is introduced to simplify the computation of Prob ($\pi_g > \pi_z$). The rationale is that if the delay of two paths are compared, the one composed of a relativly small number of hop counts will have smaller delay since the diffusion process tends to equalize the loading on all paths. The following will present an example with this concept applied to the diffusion search process.

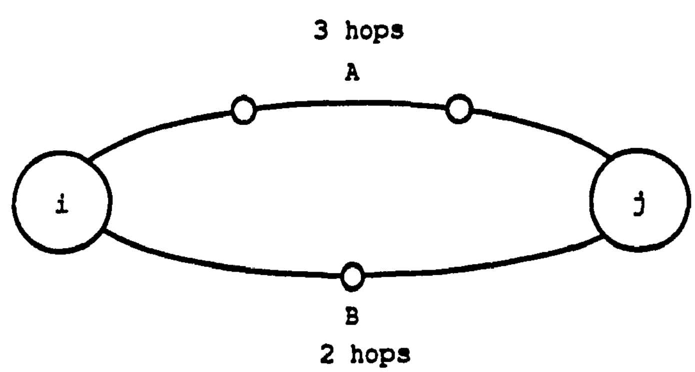

Figure 5. An example connection between switch i and j.

There are two paths connecting switch i to switch j. Path A is composed of 3 hops, and path B is composed of 2 hops. Using the hop count as the basis of comparsion, the probability that the search message originated at switch i will first reach switch j through path A will be (1 - 3/5), and for path B it will be (1 - 2/5).

Based on this idea, the set of initial values for diffusion search is generated as follows:

Hop_i : the number of hops in path i.

$$Prob_{ki}^{[r]t} = 1 - [\ Hop_t \ / \sum_{c \,\varepsilon\, Path_{ki}^{[r]}} Hop_c\ ]$$

$$\lambda_{ij}^{kr} = \sum_{\substack{c = 1 \\ c \neq j}}^{p(i)} \ \sum_{\substack{t\ \varepsilon \\ Path_{ki}^{[r]c}}} [(Prob_{ki}^{[r]t})(P_{kr})(\lambda_k)]$$

$$\lambda_{ij} = \sum_{\substack{r = 1; \\ r \neq i}}^{N} \ \sum_{\substack{k = 1; \\ k \neq i}}^{N} \lambda_{ij}^{kr} + \lambda_i$$

with 1 <= i <= N;

1 <= j <= p(i);

Once the mean arrival rate to each queue is found, the mean length of each queue can be determined by using the M/M/1 queue formula.

2.2.4 SWITCHING NETWORKS UNDER DIFFUSION ROUTING

After the search message arrives at the destination switch, a number of messages are then exchanged between the source and the destination switch. These messages are generally used to reserve the channels out of the path just traversed by the search message, and to set up a virtual circuit for the call. The problem will be made general by assuming that there will be m messages sent from the source to the destination switch, and n messages sent in the opposite direction after the search message. The message exchange is usually operated in a "send-and-wait" fashion. In other words, a switch waits to receive an acknowledge message before sending its own message(s). This kind of message exchange is assumed in the analysis. There are 3 major problems faced in applying queueing analysis when this scheme of call set-up message exchange is assumed.

(a) The first problem is caused by the bursty arrival of the call set-up messages. The interarrival time of messages to a queue depends on its distance from the corresponding source or destination switch. When it is closer to either one of them, the interarrival time of messages will become smaller. As a result, the interarrival time of an input/output queue becomes a complex function of its location in the network. The arrival process is not Poisson any more, and the analysis becomes the queueing analysis of a huge number of G/G/1 queues. The whole problem is intractable with current mathematical tools.

(b) The second problem is that the message length of the arriving messages is not uniform nor under the same distribution. Each of them has a different mean and a different distribution. As a result, the queueing analysis of each input/output queue becomes an analysis of a single queue with multiple classes of customers.

(c) The last problem relates to the problem of determining the probability of the bursty arrival of call set-up messages given the condition that a search message has arrived shortly before.

Two new ideas are proposed to solve the above 3 problems. The first one is the concept of pseudo-search messages, and the second one is the concept of traffic dependent servers.

2.2.4.1 THE FIRST MODIFICATION - PSEUDO-SEARCH MESSAGES

During diffusion search, the first search message arriving at the destination switch determines the path for call connection. Once the path for call connection is selected by the search message, all the intermediate switches of the path will receive m and n messages in opposite directions respectively, Figure 6 provides a microscopic view of switch i, which is assumed to be an intermediate switch on the selected path.

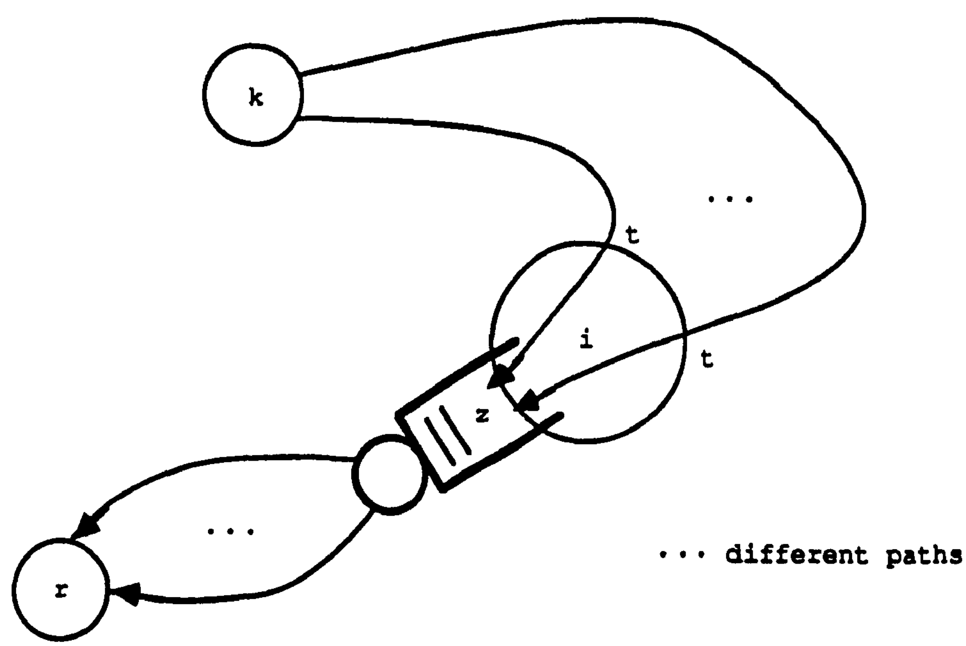

Figure 6. A microscopic view of switch i which is on a set of connected paths between the source switch k and the destination switch r.

The first search message arrives at switch i through port t, and the duplicated search message which finally results in a connection is the one sent to port z. Figure 7 provides a view on the forward and the backward messages exchanged between the source and the destination switch. If the size of the network is small or moderate and the total external arrival rate is light or moderate, the probability of messages from the other calls arriving during the interarrival time of call set-up messages is negligible. From the trunk server's point of view, serving an individual call set-up message will be similar to serving one message of a size equal to the sum of all set-up messages.

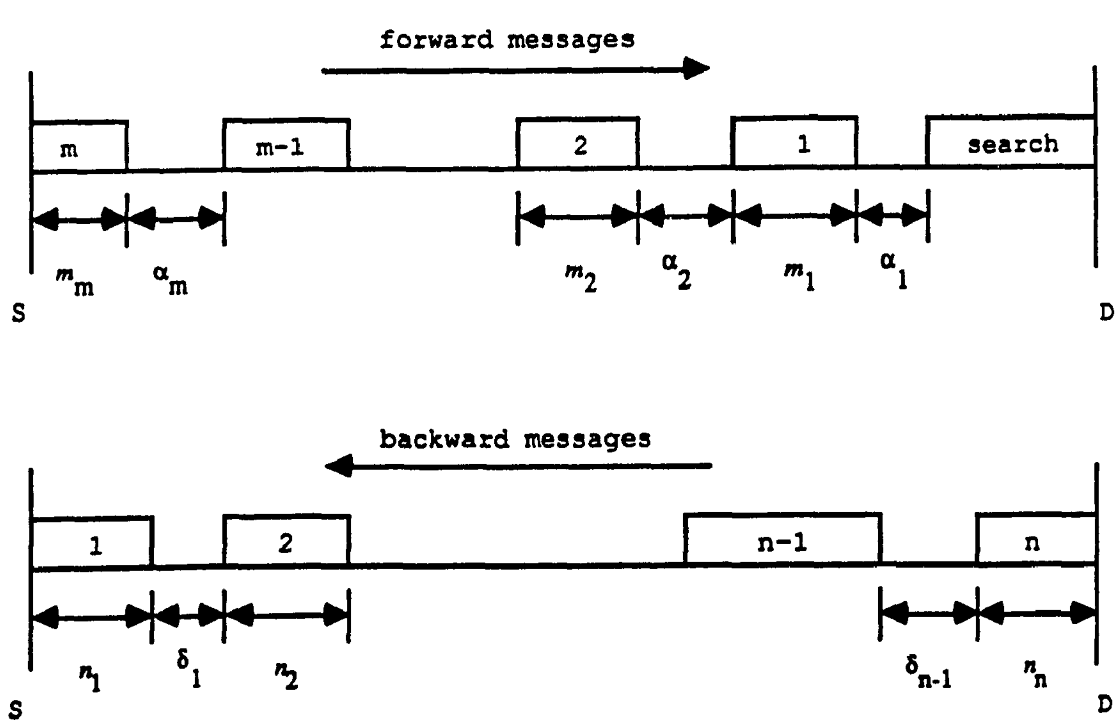

Figure 7. The forward and the backward messages exchanged between the source and the destination switch.

This expectation is justified mathematically if the probability of message arrivals due to other calls in the period $\sum m_i + \sum \alpha_i$ or $\sum n_i + \sum \delta_i$ is negligible. Let us call the port at which the search message arrived first, the arrival port. If the search message entered a switch and eventually caused it to become an intermediate switch for a call connection, a message with size equal to the sum of the n backward messages will be put into the output queue, which is associated with the arrival port. In order to approximate this backward message, a pseudo search message is put into the output queue associated with the arrival port. The probability of this event indicates the success of the arrival search message in causing a call connection. With this modification, problem (a) will be solved; the arrival process can be approximated with a Poisson process. The problem (b) is also solved partially.

2.2.4.2 THE SECOND MODIFICATION - TRAFFIC DEPENDENT SERVERS

The second modification required to extend the previous analysis of diffusion search is the concept of traffic dependent servers. Unlike the previous analysis, in which only the search message contributes traffic to the output queues, in the analysis of diffusion routing, three kinds of messages will be considered. They are

(1) search messages,

(2) search message + forward messages, and

(3) backward messages.

All the trunk servers described before are assumed to be exponential with a constant mean. In this analysis, the assumption of exponential service time is preserved, but its mean is tuned according to the incoming traffic. In other words, the mean message length of each output queue is determined based on the probability of the arrival of the search message alone, or the pseudo search message, or the search message plus the call set-up messages. In this way, there is a unique mean message length associated with each output queue of the network, and its value is dependent on its arrival process. The following terms are introduced in addition to the terms defined in the beginning to incorporate these two ideas into the analysis;

PFC_{ij}^{kr} the probability that a search message sent from switch k to switch r in the output queue j of switch i will cause the arrival of the corresponding forward message.

PBC_{ij}^{kr} the probability of putting a pseudo search message into the output queue j of switch i. This event occurs when the first search message, sent by switch k in search of switch r, enters into switch i through port j.

$1/\mu_s$ the mean service time of a search message.

$1/\mu_f$ the mean service time of a search message + n forward messages.

$1/\mu_b$ the mean service time of a search message + m backward messages.

Since the flood search in the beginning of the diffusion routing always reserves the shortest path from the source to the destination switch, PFC_{ij}^{kr} is equal to the probability that a path with output queue q_{ij} is used for connection between the source switch k and the destination switch r. Based on the principle of always finding the shortest route, it is equal to the probability that a path containing q_{ij} is the shortest out of all the possible ones connecting switch k to switch r. Using the same kind of analytical technique used in the diffusion search, PFC_{ij}^{kr} is given as:

$$PFC_{ij}^{kr} = \sum_{z \in Path_{kr}^{\{ij\}}} \prod_{g \in Path_{kr}^{[ij]}} Pr(\pi_g >= \pi_z)$$

The rationale of the above formula is that PFC_{ij}^{kr} is equal to the probability that the path with the smallest path delay belongs to $Path_{kr}^{\{ij\}}$. The mathematical expression of $Pr(\pi_z >= \pi_g)$ is the same as the one shown in the diffusion search. The probability of the arrival of the call set-up messages to the output queue j of switch i given the condition that a search message arrived shortly before is

$$PFC_{ij}^{*} = \sum_{\substack{k = 1; \\ k \neq i}}^{N} \sum_{\substack{r = 1; \\ r \neq i}}^{N} PFC_{ij}^{kr}$$

The above value of PFC_{ij}^{*} represents the probability limited to the search messages not generated by switch i itself. The PFC_{ij} contribution due to the exogenous traffic entering switch i is computed in the following:

$$PFC_{ij}^{\#} = \sum_{\substack{r = 1; \\ r \neq i}}^{N} \sum_{z \in Path_{ir}^{\{ij\}}} \prod_{g \in Path_{ir}^{[ij]}} Pr(\pi_g >= \pi_z)$$

in other words,

$$PFC^{\#}_{ij} = \sum_{r \neq i} PFC^{ir}_{ij}$$

The rationale of computing $PFC^{\#}_{ij}$ is the same as before, except that source switch is switch i, and the destination switch is selected from the rest of the switches of the network. Thus

$$PFC_{ij} = \sum_{k=1;}^{N} \sum_{\substack{r=1; \\ r \neq i}}^{N} \sum_{z \,\varepsilon\, Path^{\{ij\}}_{kr}} \prod_{g \,\varepsilon\, Path^{[ij]}_{kr}} Pr(\pi_g >= \pi_z)$$

$$= \sum_{k=1}^{N} \sum_{\substack{r=1 \\ r \neq i}}^{N} PFC^{kr}_{ij}$$

With respect to network topology, if two switches are connected, an operator CONNECT(i, j) is defined to locate the connected output queue of the output queue j of switch i.

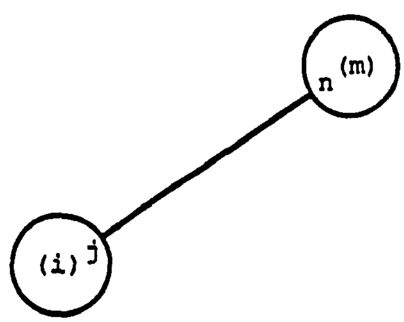

Figure 8. The graphical illustration of the CONNECT(i,j) operator.

With reference to Figure 8,

$$PBC_{ij}^{kr} = PFC_{CONNECT(i,\ j)}^{kr}, \text{ and } CONNECT(i,\ j) = \langle m,\ n \rangle$$

It is obvious that the probability PBC_{ij} contributed by switch i itself is equal to zero. Then the probability of putting a pseudo search message in the output queue in which the first search message arrives is equal to

$$PBC_{ij}^{kr} = \sum_{\substack{r = 1; \\ r \neq i}}^{N} \sum_{\substack{k = 1; \\ k \neq i}}^{N} PFC_{CONNECT(i,\ j)}^{kr}$$

The concept behind the traffic dependent server is that the mean service time of the trunk server should reflect the message proportion of the arrival traffic. The mean service time SER_{ij} associated with output queue j of switch i, is tuned with respect to the probability of the arrival of the search message alone, or the search message plus the call set-up messages, or the pseudo search message. Then

$$SER_{ij} = PFC_{ij}/\ \mu_f + PBC_{ij}/\ \mu_b + (1 - PFC_{ij} - PBC_{ij})/\ \mu_s$$

The mean arrival rate λ_{ij} to the output queue j of switch i which is modified to incorporate the backward messages and the forward messages is given below:

$$\lambda_{ij} = \lambda_i + \sum_{\substack{k=1;\\ k\neq i}}^{N} \sum_{\substack{r=1;\\ r\neq i}}^{N} [\, \lambda_{ij}^{kr} + (PBC_{ij}^{kr})(P_{kr})(\lambda_k)\,]$$

Using the above definition of SER_{ij} and λ_{ij}, the traffic intensity factor ρ_{ij} is

$$\rho_{ij} = \lambda_{ij} / SER_{ij}$$

As in the previous analysis of the diffusion search, the above expression for ρ_{ij} applies to every output queue of the network with the assumption that the traffic intensity factor of all the output queues are known, except ρ_{ij}. A set of recursive equations can thus be generated. By solving this set of equations iteratively, the traffic intensity factor of all the output queues can be found. Once the traffic intensity factor of a queue is found, all the characteristics associated with this queue can be determined using the M/M/1 queue formulas. The details of these formula are described in many standard books Kleinrock[2] Hayes[4] on queueing theory. In concluding this section, the following set of equations has to be solved simultaneously in order to find the traffic intensity factor of each input/output queue of the network.

For a network with N switches, and each arbitrary switch i of the network has p(i) input/output queues, the following $\sum_{i=1}^{N} p(i)$ sets of equations have to be solved simultaneously.

$$Pr(\pi_z >= \pi_t) =$$

$$= \sum_{k \in \pi_z} \sum_{l \in \pi_t} [(\sigma_k^{\pi_z} \cdot \sigma_l^{\pi_t})(\mu - \lambda_l)] / [(\mu - \lambda_l)(\mu - \lambda_k)(2\mu - \lambda_l - \lambda_k)]$$

$$\text{with } \sigma_i^{\pi_r} = [\prod_{k \in \pi_r} (\mu - \lambda_k)] / [\prod_{k \in \pi_r;\, k \neq i} (\lambda_i - \lambda_k)]$$

$$PFC_{ij} = \sum_{k=1;\, k \neq i}^{N} \sum_{r=1;\, r \neq i}^{N} \sum_{z \in Path_{kr}^{\{ij\}}} \prod_{g \in Path_{kr}^{[ij]}} Pr(\pi_g >= \pi_z) +$$

$$\sum_{r=1;\, r \neq i}^{N} \sum_{z \in Path_{kr}^{\{ij\}}} \prod_{g \in Path_{kr}^{[ij]}} Pr(\pi_g >= \pi_z)$$

$$PBC_{ij}^{kr} = \sum_{\substack{r=1;\\ r \neq i}}^{N} \sum_{\substack{k=1;\\ k \neq i}}^{N} PFC^{kr}_{CONNECT(i,\ j)}$$

$$SER_{ij} = PFC_{ij} / \mu_f + PBC_{ij} / \mu_b + (1 - PFC_{ij} - PBC_{ij}) / \mu_s$$

$$\lambda_{ij} = \lambda_i + \sum_{\substack{k=1;\\ k \neq i}}^{N} \sum_{\substack{r=1;\\ r \neq i}}^{N} [\ \lambda_{ij}^{kr} + (PBC_{ij}^{kr})(P_{kr})(\lambda_k)]$$

$$\rho_{ij} = \lambda_{ij} / SER_{ij}$$

$$1 <= i <= N;\ 1 <= j <= p(i)$$

The set of initial values for the above set of equations is generated using the same concept as in the diffusion search. In computing the initial values, the value of PFC_{ij} is calculated using the following formula,

$$1_{kr}^{ij}(z) = 1 - Hop_z / [\ \sum_{g\ \varepsilon\ Path_{kr}^{[ij]}} Hop_g + Hop_z\ ]$$

$$PFC_{ij} = \sum_{\substack{k=1;\\ k\neq i}}^{N} \sum_{\substack{r=1;\\ r\neq i}}^{N} \sum_{\substack{z\,\varepsilon\\ Path_{kr}^{\{ij\}}}} 1_{kr}^{ij}(z) \; + $$

$$\sum_{\substack{r=1;\\ r\neq i}}^{N} \sum_{\substack{z\,\varepsilon\\ Path_{ir}^{\{ij\}}}} 1_{ir}^{ij}(z);$$

The rationale of the above formula is to use hop count instead of queueing delay to calculate PFC_{ij}, which is the sum of PFC_{ij}^{kr} with different combinations of k and r. PFC_{ij}^{kr} is the probability that a search message sent from switch k to switch r in the output queue j of switch i will cause the arrival of the corresponding forward message. In other words, it is equal to the probability that the path traversed by the search message sent from switch k to switch r composed of output queue j of switch i is the shortest one in delay out of all the possible ones. Since hop count is used instead of queueing delay, this approximation is topology dependent only, and the input traffic to the switching networks is not taken into account.

CHAPTER III

COMPARSION BETWEEN SIMULATION AND ANALYTICAL RESULTS

The objective of this section is to compare the simulation results with the analytical results derived in the previous chapter.

3.1 SIMULATION OUTLINE

The simulation was done using the SAMOC[14] package developed at the University of Calgary. The network being studied with the simulation package is shown Figure 9.

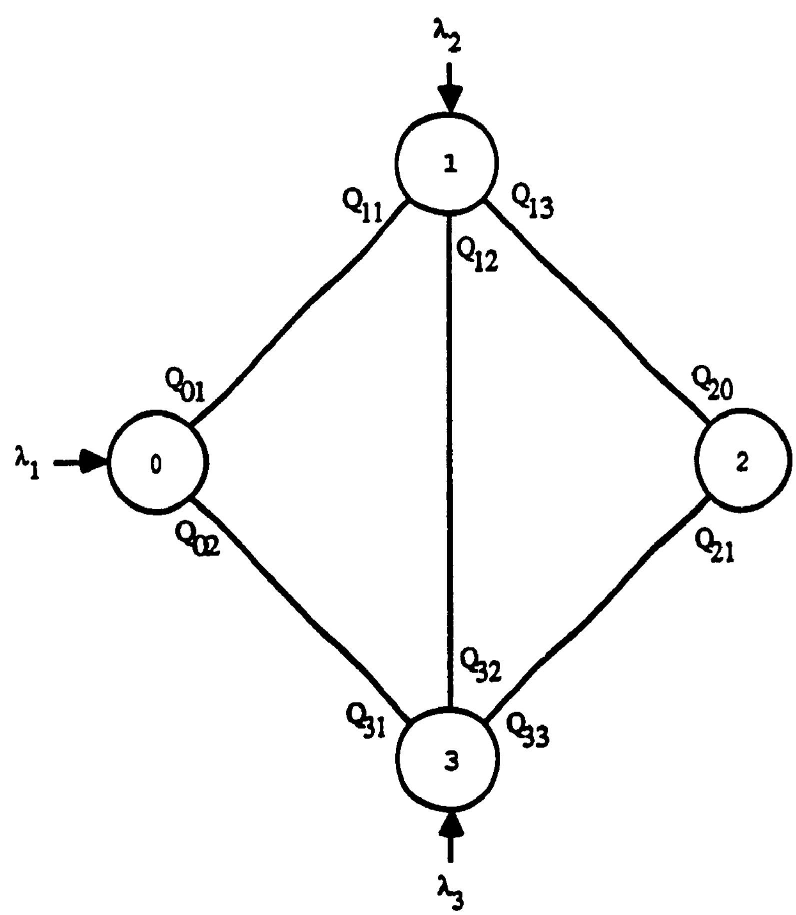

Figure 9. The network being studied by the simulation

It is composed of four switches and three traffic generators. The switch is used to model the action of the switch described in Section 2.2.2, and the traffic generator is used to model the action of the local subscribers attached to the switch. Local calls are not considered in the simulation. Calls are generated with a Poisson distribution. For each call, the destination switch is selected with equal probability from the other switches in the network. Obviously, a switch can only be eligible to be a destination switch if it is attached to a traffic generator. Q_{ij} denotes the output queue j of switch i and ρ_{ij} is the traffic intensity factor associated with queue Q_{ij}.

3.2 DIFFUSION SEARCH SIMULATION

The following tables list the simulation and the analytical results of diffusion search with different external arrival rates. The service time (packet length) of the trunk server is assumed to be exponentially distributed with mean equal to 40. λ_1, λ_2, and λ_3 are the external arrival rates generated by the traffic generators 1, 2, and 3, respectively.

Table 1

Simulation time: 8000 seconds

$\lambda_1 = 0.4$ $\lambda_2 = 0.4$ $\lambda_3 = 0.4$

	Simulation result	Analytical result	Difference (%)
ρ_{01}	0.015959	0.015263	4.36
ρ_{02}	0.014581	0.015097	3.53
ρ_{11}	0.014937	0.015016	0.52
ρ_{12}	0.017225	0.016251	5.65
ρ_{13}	0.019089	0.018110	5.13
ρ_{20}	0.013979	0.015263	9.18
ρ_{21}	0.015346	0.015015	2.15
ρ_{31}	0.015310	0.015181	0.842
ρ_{32}	0.014846	0.016542	11.42
ρ_{33}	0.018916	0.019018	0.539

Table 2

Simulation time: 8000 seconds

$\lambda_1 = 0.8$ $\lambda_2 = 0.4$ $\lambda_3 = 0.8$

	Simulation result	Analytical result	Difference (%)
ρ_{01}	0.031490	0.030649	2.6
ρ_{02}	0.025564	0.025455	0.426
ρ_{11}	0.019644	0.020044	2.036
ρ_{12}	0.024298	0.022688	6.626
ρ_{13}	0.027606	0.027760	0.5578
ρ_{20}	0.027952	0.029229	4.568
ρ_{21}	0.022530	0.021464	4.73
ρ_{31}	0.025894	0.025237	2.537
ρ_{32}	0.031984	0.031775	0.653
ρ_{33}	0.034785	0.034255	1.5[illegible]

Based on the tables shown on the previous pages, there are discrepancies between the simulation and the analytical results. The causes of these discrepancies are explained in the following:

. In Table 1, the external arrival rates are symmetrical to the network configuration - the arrival rate to switch 1 and 3 is the same. The following symmetrical results should be expected:

$\rho_{01} = \rho_{02}$;

$\rho_{11} = \rho_{31}$;

$\rho_{12} = \rho_{32}$;

$\rho_{13} = \rho_{33}$;

$\rho_{20} = \rho_{21}$;

With respect to the expected results shown above, there is a general discrepancy existing among the simulation results. The difference is within 0.0001 in most of the cases and the most serious one is between queue Q_{12} and Q_{32}, the difference is 0.002379. These inconsistencies are merely caused by the limited confidence level of the simulation. In other words, the results of the simulation have not settled down yet, and some parts of the network are still governed by the transient behavior. It is my belief that the simulation results will eventually settle down when the simulation time is long enough. The problem of estimating confidence interval has already been identified and discussed in the introduction to this thesis. This is because the percentage of the confidence level of the simulation results with respect to the simulation time is not yet known. In order to gain a feeling of the confidence level of simulation outputs, the mean arrival rate to each output queue derived analytically, is put into the formula shown in APPENDIX C. These formulas provide means to estimate the confidence level of the simulation output. The confidence level of the diffusion search simulation is approximately between 0.9 and 0.95. Based on the method shown in APPENDIX D, the transient period of the simulation is found to be negligible. All the models enters into the steady state after the first 100 seconds of simulation.

. The expected symmetry found in the analytical results is within a difference of 0.0003 in most cases, and the most serious case is found between the queues Q_{13} and Q_{33}, in which the difference is 0.0009. The differences is believed to be caused by floating point truncation when iterating the set of nonlinear equations. In terms of the expected symmetry, the analytical results provide more confidence than the simulation results.

. In Table 1, the external arrival rate is not uniform across all the switches in the network, there is no arrival to switch 2 and the arrival rate to switch 0 is bigger than switches 1 and 3. As a result, the following phenomena is observed:

$\rho_{33} > \rho_{32} > \rho_{31}$; and

$\rho_{13} > \rho_{12} > \rho_{11}$;

The reason behind the above phenomena is the operation of the diffusion search. For an intermediate switch, the arrival rate to an output queue associated with port i is equal to the sum of arrival rates to ports other than i during the diffusion search. Along this line of reasoning, the arrival rate to the output queue, Q_{33}, is equal the sum of the following arrivals:

(1.) λ_2 arriving through the paths 1 - 3 and 1 - 0 - 3

(2.) λ_1 arriving through the paths 0 - 3 and 0 - 1 - 3

(3.) λ_3

While the arrival rate to the output queue, Q_{32}, is equal to the sum of the following arrivals:

(1.) λ_2 arriving through the paths 1 - 0 - 3 and 1 - 2 - 3

(2.) λ_1 arriving through the paths 0 - 3 and 0 - 1 - 3

(3.) λ_3

Since the delay along the path 1 - 3 (1 hop) is more likely shorter than the path 1 - 2 - 3 (2 hops), the arrival λ_2 is more likely to select path 1 - 3 to reach switch 3 than the other path. As a result, we should expect $\rho_{33} > \rho_{32}$. Based on this line of reasoning, the above phenomena can be explained.

- In Table 2, the external arrival rates are asymmetrical with respect to the network configuration. For example, the arrival rate to switch 3 is different from the one to switch 1. Asymmetrical results are expected and are found both in the analytical and the simulation results.

- Based on these tables, the accuracy of the analytical results are within 10 percent of the simulation results. This is mainly because the service time of the trunk server used by the simulation is exponentially distributed, and as a result Kleinrock's independence assumption can be applied directly.

Tables 3, 4, 5, and 6 list the simulation and the analytical results of the diffusion protocol for different external arrival rates. The number of forward messages and the backward messages are equal to 2 and 3 respectively. The service time of all the forward messages and all the backward messages are 60 and 30. The entry of "******" under the Difference column of the tables means that the queueing analysis derived in Section 2.2.4 is not applicable, and the reason will be elaborated at the end of this section .

3.3 DIFFUSION ROUTING SIMULATION

The following lists the simulation and the analytical results:

Table 3

Simulation time: 8000 seconds

$\lambda_1 = 0.8$ $\lambda_2 = 1.5$ $\lambda_3 = 0.8$

	Simulation result	Analytical result	Difference (%)
ρ_{01}	0.045519	0.047485	4.319
ρ_{02}	0.042297	0.049447	16.90
ρ_{11}	0.059110	0.053653	9.23
ρ_{12}	0.059363	0.055737	6.10
ρ_{13}	0.043128	0.058644	*****
ρ_{20}	0.000000	0.036028	*****
ρ_{21}	0.001000	0.048363	*****
ρ_{31}	0.041976	0.048252	14.95
ρ_{32}	0.044710	0.050957	13.97
ρ_{33}	0.033690	0.048718	*****

Table 4

Simulation time: 8000 seconds

$\lambda_1 = 0.8 \quad \lambda_2 = 0.35 \quad \lambda_3 = 0.08$

	Simulation result	Analytical result	Difference (%)
ρ_{01}	0.036538	0.037692	3.158
ρ_{02}	0.038092	0.037731	0.949
ρ_{11}	0.036932	0.037645	1.93
ρ_{12}	0.036249	0.039320	8.472
ρ_{13}	0.027322	0.039304	******
ρ_{20}	0.001000	0.031797	******
ρ_{21}	0.001000	0.031643	******
ρ_{31}	0.036702	0.037748	2.849
ρ_{32}	0.036810	0.039325	6.832
ρ_{33}	0.026695	0.039492	******

Table 5

Simulation time: 8000 seconds

$\lambda_1 = 1.5$ $\lambda_2 = 0.8$ $\lambda_3 = 0.8$

	Simulation result	Analytical result	Difference (%)
ρ_{01}	0.058881	0.054177	8.00
ρ_{02}	0.060374	0.053571	11.268
ρ_{11}	0.043692	0.048151	10.20
ρ_{12}	0.042708	0.050545	18.35
ρ_{13}	0.032680	0.049683	*****
ρ_{20}	0.010000	0.042145	*****
ρ_{21}	0.000000	0.042173	*****
ρ_{31}	0.043301	0.048583	12.198
ρ_{32}	0.042895	0.050571	17.89
ρ_{33}	0.032342	0.049820	*****

Table 6

Simulation time: 8000 seconds

$\lambda_1 = 0.15$ $\lambda_2 = 0.15$ $\lambda_3 = 0.20$

	Simulation result	Analytical result	Difference (%)
ρ_{01}	0.008415	0.008067	4.135
ρ_{02}	0.008033	0.007881	1.89
ρ_{11}	0.007219	0.007915	9.64
ρ_{12}	0.007552	0.008308	10.0
ρ_{13}	0.006210	0.008210	*****
ρ_{20}	0.000000	0.007032	*****
ρ_{21}	0.000000	0.006416	*****
ρ_{31}	0.007342	0.008174	11.33
ρ_{32}	0.007979	0.008532	6.93
ρ_{33}	0.005713	0.008677	*****

Based on these tables, the range of discrepancy differs depending on the external arrival rate, and the position of the output queue in the network. These relationships will be elaborated upon in the following:

. A trend of discrepancy between the simulation and the analytical results with respect to the external arrival rate is observed in Tables 3, 4, 5 and 6. The discrepancy increases as the external arrival rate is increased. This phenomenon is a result of the collapse of one of the main approximations used in the analysis of the diffusion routing. When the arrival rate increases, the probability of a message transmission generated by one call interrupting the message transmission of other calls is no longer negligible.

. A large discrepancy exists between the simulation and the analytical results in the output queues Q_{20}, Q_{21}, Q_{33}, and Q_{13}. This big discrepancy is caused by the failure of applying the M/M/1 queue analysis to these output queues and the diminishing confidence of the simulation. With respect to the network topology, the probability of a call established through these output queues is very low. As a result, the message arrival rate to these output queues is also low, decreasing the accuracy of the simulation outputs associated with these output queues. Also, the messages that arrive at these output queues are mainly search messages because these queues are rarely used for call connection. The analytical solution estimates that more than 90 percent of the traffic arriving to these queues consists of search messages. Hence, these output queues behave more like M/D/1 queues than M/M/1 queues, thereby making the M/M/1 queue approach invalid.

3.4 OUSTANDING PROBLEMS

In concluding this chapter, the following will list the outstanding problems related to the study of diffusion routing. These problems cover the computational aspect and the mathematical nature of the diffusion routing. Besides pointing out the shortcomings of the analysis just given, they provide future directions for research in this area.

- The analysis requires exhaustive enumeration of all the possible paths in the network. It will become computationally intractable when studying practical networks, which range from 10 to 100 switches. Therefore, an efficient algorithm for enumerating all the distinct paths in a network should be investigated in order to place the above analysis on a practical ground.

- With respect to the equations shown in APPENDIX B, the mathematical expression derived for the path delay comparison between two paths is valid if and only if the two paths are distinct. In other words, these two paths should not have output queues with the same arrival rates. If this case should arise, two solutions are available to continue the analysis. The first solution is to remove the output queue having the same arrival rate from both of the paths. The other is to modify the partial fraction used in APPENDIX B to generate a new expression for path delay comparison. However, the arrival rate to each output queue in the network changes while iterating the set of nonlinear equations. Therefore, the problem on how to incorporate the above modification when iterating the equations should be investigated.

- The simulation technique is crucial in studying the diffusion routing. The analysis of diffusion routing requires extensive simulation to validate. Therefore, efficient simulation techniques should be developed in order to gain a deeper understanding. The distributed simulation technique seems to be promising in studying this kind of super large system and it is currently under investigation.

- The analysis involves solving a large system of nonlinear equations. There is a set of four nonlinear equations associated with each output queue in the network. For a network of 10 switches, the analysis will be required to solve about four hundred simultaneous nonlinear equations. Obviously, an efficient algorithm for solving a large system of nonlinear equations is essential to the analysis, and a study of this is currently underway.

- The analysis of diffusion routing rests on an assumption that the network will eventually converge to an equilibrium point. However, the analysis generates a large set of nonlinear equations and opens the possibility that more than one equilibrium point may exist in the network. The original assumption of a single equilibrium point requires more justification.

- In the section comparing the simulation results with the analytical results, discrepancy is observed at the output queue which does not have enough mixed traffic. The reason behind the discrepancy is that Kleinrock's independent assumption does not hold. Therefore, ways have to be sought to relax Kleinrock's assumption when studying output queues with little mixing of traffic. The diffusion technique described in Gelenbe[7] for modelling switching network seems to be promising.

. When the external arrival rate to the network increases, general discrepancies increase between the simulation and the analytical results. The cause of this is due to the fact that the traffic dependent server approximation fails when the arrival rates from different sources increase.

. It is instructive to estimate the overhead of diffusion routing originated by the diffusion search. The overhead means the amount of traffic generated which is not involved in setting up the virtual circuit. Having the knowledge of the overhead, the throughput of diffusion routing can be determined, and the efficiency of this technique can be appreciated.

. The way the network collapses when the external arrival rate increases without bound is an important behavior to understand. When an output queue collapses, its traffic intensity factor becomes 1 and the queueing delay becomes infinity. Is it possible that a small part of the network collapses, and the rest of the network still survives? Or, when a small part of the network collapses, will the failure propagate to the whole network eventually? These questions are important in understanding the diffusion routing, and must be investigated in greater detail.

CHAPTER IV

APPLICATIONS

4.1 LIST OF POSSIBLE APPLICATIONS

The goal of the following section is to illustrate examples where the results derived in the previous sections are applied to solve some practical network problems. When a military network is planned, there are two major problems faced by the modern network planner. The first one is the call set-up time. Call set-up time is the time from the moment the number of the called subscriber is dialled in until the moment the phone of the called subscriber rings. This time is very important in military communication, and many lives may be affected if the call set-up time takes more than a few seconds. The next problem is the mean trunk utilization pattern. The mean trunk utilization pattern indicates the mean number of channels used in each trunk of the network. If all the channels of a trunk are used, that trunk will block all arriving calls. In planning a military network, it is crucial to predict which trunk of the network will block arriving calls and what remedy can be used before the action starts. If all the trunks connected to a switch are fully utilized, this switch will have problems communicating with the outside world.

4.2 THE MEAN CALL SET-UP TIME

In light of the previous analysis, call set-up time is the time required to set up a virtual circuit between the source and the destination switches. This time is measured from the moment the source starts the diffusion search until the moment it receives the last backward message. Therefore, if path π_k is taken to set up the virtual circuit, the call set-up time is equal to the sum of the queueing time at each output queue and the transmission time of each trunk along π_k.

$$t_{\text{call set-up}} = \sum_{\pi_k} [(\text{queueing time}) + (\text{transmission time})]$$

Using the same nomenclature developed in the previous section, the queueing time of a message in an output queue depends on the traffic intensity factor associated with that queue, and is given as (for output queue j of switch i, q_{ij})

$$\text{queueing time}_{q_{ij}} = [2\lambda_{ij}(1 - \rho_{ij})] \,/\, [SER_{ij}(1 - \rho_{ij})]^2$$

(the derivation of this formula is shown in APPENDIX B)

$$\text{transmission time}_{q_{ij}} = \text{message length} \,/\, SER_{ij}$$

Then, the mean call set-up time between the source switch k and the destination switch r is equal to the sum of the mean time to send n forward messages along the forward path plus the mean time to send n messages along the backward path.

$$= \sum_{\substack{\pi_k \ \varepsilon \\ Path_{kr}}} \Big[\sum_{(i,\ j)\ \varepsilon\ \pi_k} \{ [2\lambda_{ij}(1 - \rho_{ij})] \ / \ [SER_{ij}(1 - \rho_{ij})^2] +$$

$$[2\lambda_{st}(1 - \rho_{st})] \ / \ [SER_{st}(1 - \rho_{st})^2] +$$

$$\mu_f \ / \ SER_{ij} + \mu_b \ / \ SER_{st} \} \Big]$$

with st = CONNECT(i, j)

In a network of N switches, N x N-1 mean call set-up times can be defined using the above formula. Each one corresponds to a distinct source and destination switch pair.

4.3 THE MEAN TRUNK UTILIZATION PATTERN

The mean trunk utilization pattern indicates the mean number of channels used in each trunk of the network for call connection. Although the number of channels of each trunk is assumed to be unlimited during the derivation process of the traffic intensity factor, the results related to the probability PFC can be used to study the mean number of channels used inside a trunk, throughout the life time of a network. If the mean number of channels used is greater than the number available in that trunk, then trunk blocking occurs. The following iterative procedures are defined to derive the trunk utilization pattern of a network based on the values of the PFCs.

c_{ij} the maximum number of channels available on the trunk with one side connected to the output queue j of switch i.

t_{ij} the mean number of trunk channels used with one side connected to the output queue j of switch i throughout the life time of the network.

The definition of PFC_{ij}^{kr} is the probability that the search message sent from switch k to switch r via output queue j of switch i, will cause the arrival of the corresponding forward message. In other words, PFC_{ij}^{kr} is equal to probability that the source switch k uses a virtual circuit, composed of the trunk with one side connected to the output queue j of switch i, to connect to the destination switch r. Therefore t_{ij} can be determined by

$$t_{ij} = \sum_{\substack{k=1;\\ k\neq i}}^{N} \sum_{\substack{r=1;\\ r\neq i}}^{N} [(PFC_{ij}^{kr})(\lambda_{kr})] + \sum_{\substack{r=1;\\ r\neq i}}^{N} [(PFC_{ij}^{ir})(\lambda_{ir})]$$

The iterative procedure used to determine the trunk utilization pattern is described in the following.

Initialization:

Use the set of equations derived in Section 2.2.4 to generate the PFC_{ij}^{kr} of each output queue in the network. Then initialize t_{ij} using the above formula.

Iteration:

while at least one t_{ij} out of all the trunks is greater than its associated c_{ij} and the switching network is still connected, then do

begin

for each t_{ij} greater than its associated c_{ij}, do

begin

for each switch k of the network, do

begin

$$\lambda_k = \lambda_k - \left(\sum_{\substack{r = 1; \\ r \neq i}}^{N} [(PFC_{ij}^{kr})(\lambda_{kr})]\right)(c_{ij}/ t_{ij});$$

end;

remove the associated trunk (blocking trunk) from the network;

end;

use the set of equations derived in Section 2.2.4 to generate the PFC_{ij}^{kr} of each output queue of the network;

t_{ij} =

$$\sum_{\substack{k = 1; \\ k \neq i}}^{N} \sum_{\substack{r = 1; \\ r \neq i}}^{N} [(PFC_{ij}^{kr})(\lambda_{kr})] + \sum_{\substack{r = 1: \\ r \neq i}}^{N} [(PFC_{ij}^{ir})(\lambda_{ir})]$$

end;

The rationale of the above iterative procedure is based on the diffusion automatically bypassing a blocked trunk and setting up the virtual circuit through other trunks as long as the network is connected. Therefore, a blocked trunk will have no effect on the diffusion search except that it provides a share of channels statistically to all the switches in the network.

CHAPTER V

CONCLUSION

This thesis has described a modelling technique for a switching network operating under diffusion routing. In order to make the problem mathematically tractable, the performance analysis is limited to mean value analysis, and many approximations are introduced. However, the analytical results are very promising. In the study of diffusion search, all the analytical results are within ten percent of the simulation results and typically much better. In the diffusion routing, only some of the output queues can have results as good as in the diffusion search depending on their locations and the external arrival rate to the whole network. Although there are a few important problems still outstanding and waiting to be resolved, this analysis represents one of the first attempts to stretch queueing theory in order to gain an understanding of complex routing techniques such as diffusion.

REFERENCES

[1] L. Kleinrock, Communication Nets - Stochastic Message Flow and Delays, McGraw-Hill, New York (1964).

[2] L. Kleinrock, Queueing System, Vol I : Theory, John Wiley, New York (1975).

[3] L. Kleinrock, Queueing System, Vol II : Computer Application, John Wiley, New York (1976).

[4] Hayes. Jeremiah F., Modeling and Analysis of Computer Communications Networks, Plenum Press, New York (1984).

[5] William Stallings, Data and Computer Communications, Macmillian, New York (1985).

[6] A. S. Tannenbaum, Computer Networks, Prentice-Hall, Englewood Cliffs, New Jersey (1981).

[7] E. Gelenbe and I. Mitrani, Analysis and Synthesis of Computer Systems, Academic Press (1980).

[8] H. Rudin, "On Routing and Delta Routing a Taxonomy and Performance Comparsion for Packet Switched Network," IEEE Transactions on Communications, Com-24(1), 43-59, January (1976).

[9] J. W. Wong and S. S. Lam, "Queueing Network Models of Packet Switching Networks: Part 1: Open Networks, Performance Evolution, Vol 2 (1982).

[10] J. W. Wong, "Distribution of end-to-end delay in message switched networks", Computer Networks 2 (1978).

[11] J. W. Wong, "Distribution of end-to-end delay in packet switching networks - Exact and approximate results, CCNG Rept., University of Waterloo.

[12] M. Schwartz and T. E. Stern, "Routing Techniques used in Computer Communication Networks," IEEE Transactions on Communications, Com-28 (4), 539-553, April (1980).

[13] J. R. Jackson, "Jobshop-like queueing systems," Management Sci. 10 (1963).

[14] G. Lomow, B. Unger, and G. Birtwistle, SIMDC User's Reference Manual, University of Calgary (1983).

[15] M. H. MacDougall, Simulating Computer Systems, Techniques and Tools, MIT Press (1987).

[16] S. S. Lavenberg, Computer Performance Modeling Handbook, Academic Press (1983).

[17] P. E. Cantrell, "Computation of the Transient M/M/1 Queue cdf, pdf, and Mean with Generalized Q-Function," IEEE Transactions on Communications, Com-34 (8), 814-817, August (1986).

[18] A. M. Law, "Statistical Analysis of Simulation Output Data," Operations Research 31, 6 (1983), 983-1027.

APPENDIX A

PATH DELAY FORMULA

When the nonpassing condition is valid, the Laplace transform of the end-to-end path delay distribution is given as

$$T_k^*(s) = \prod_{i \in \pi_k} [(\mu_i(1 - \rho_i)) / (s + (\mu_i(1 - \rho_i))]$$

By inverse Laplace transform and partial fraction, the end-to-end path delay distribution in time domain is

$$t_k(t) = \sum_{i \in \pi_k} \sigma_i^{\pi_k} \exp(-\sigma_i^{\pi_k} (1 - \rho_i))$$

and

$$\sigma_i^{\pi_k} = [\prod_{k \in \pi_k} (\mu - \lambda_k)] / [\prod_{k \in \pi_k;\, k \neq i} (\lambda_i - \lambda_k)]$$

The above results are derived based on the assumption that no two λ_i have the same value. If it is not the case (these two paths are correlated), the derivation process will be more complex but is still mathematically tractable. In order to derive the mathematics of the expression, $Pr(\pi_r >= \pi_s)$, two paths ,π_r and π_s, are assumed with delay characterized by the delay function $t_r(t)$ and $t_s(t)$, and these two paths are assumed to be disjoint (they do not have the same set of traffic intensity factors or arrival rates). The probability of the event that $t_r(t) >= t_s(t)$ will be derived in the following:

$$t_r(t) = \sum_{i \varepsilon \pi_r} \sigma_i^{\pi_r} \exp(-\mu_i(1 - \rho_i)) \text{ and}$$

$$t_s(t) = \sum_{j \varepsilon \pi_s} \sigma_j^{\pi_s} \exp(-\mu_j(1 - \rho_j)) \text{ and}$$

$$Pr(t_r(t) >= t_s(t)) =$$

$$\int_0^\infty \int_0^r [\sum_{i \varepsilon \pi_r} \sigma_i^{\pi_r} \exp(-\mu_i(1 - \rho_i))][\sum_{j \varepsilon \pi_s} \sigma_j^{\pi_s} \exp(-\mu_j(1 - \rho_j))]drds$$

By manipulating the integrals, it is equal to

$$= \int_0^\infty \sum_{i \varepsilon \pi_r} \sigma_i^{\pi_r} \exp(-\mu_i(1 - \rho_i))dr \int_0^r \sum_{j \varepsilon \pi_s} \sigma_j^{\pi_s} \exp(-\mu_j(1 - \rho_j))ds$$

$$= \int_0^\infty \Big[\sum_{i \,\varepsilon\, \pi_r} \sigma_i^{\pi_r} \exp(-\mu_i(1 - \rho_i)) \Big]$$

$$\Big[\sum_{j \,\varepsilon\, \pi_s} \{ \sigma_j^{\pi_s} (1 / \mu_j(1 - \rho_j)) -$$

$$(\exp(-\mu_j(1 - \rho_j)) / (\mu_j (1 - \rho_j))) \} \Big]$$

After some manipulations, the final forms of $\Pr(t_r(t) >= t_s(t))$ is given as

$$\sum_{i \,\varepsilon\, \pi_r} \sum_{j \,\varepsilon\, \pi_s} [(\sigma_i^{\pi_r} . \sigma_j^{\pi_s})(\mu_j - \lambda_j)] / [(\mu_j - \lambda_j)(\mu_i - \lambda_i)(\mu_j + \mu_i - \lambda_j - \lambda_i)]$$

APPENDIX B

M/M/1 QUEUEING TIME

The distribution of the queueing time of a M/M/1 queue experienced by a message is given as

$$q(t) = \delta(t)(1 - \rho) + \lambda(1 - \rho)\exp(-\mu(1 - \rho)t)$$

The mean queueing time is obtained by taking the expectation of q(t),

$$q^* = \int_0^\infty [\ \delta(t)(1 - \rho) + \lambda(1 - \rho)\exp(-\mu(1 - \rho)t)\ ]t\ dt$$

and the result is equal to

$$q^* = [\ \lambda(1 - \rho)\Gamma(2)] \ /\ [\mu(1 - \rho)]^2$$

APPENDIX C

M/M/1 QUEUE TRANSIENT TIME

The following lists the generalized Q-function expressions developed in Cantrell[17] for the transient state occupancy cumulative distribution function (cdf) for an M/M/1 queue.

To derive a generalized Q-function representation for the cdf $F_t(n) = \Pr\{n_t <= n \mid n_{t0} = n_0\}$ with n_0 customers were initially present at t = t0, (in our case, $n_0 = 0$ and t0 = 0.) let $\alpha = (2\rho\mu t)^{1/2}$, $\beta = (2\mu t)^{1/2}$,

$$F_t(n) = -\rho^{n+1}(1 - Q_{n+n_0+2}(\alpha, \beta))$$

$$+ Q_{n-n_0+1}(\beta, \alpha)$$

Two well-known series expansions for the generalized Q-function are

$$1 - Q_m(\alpha, \beta) = \exp(-(\alpha^2 + \beta^2)/2) \sum_{k=m}^{\infty} (\beta/\alpha)^k I_k(\alpha\beta)$$

and

$$Q_m(\alpha, \beta) = \exp(-(\alpha^2 + \beta^2)/2) \sum_{k=1-m}^{\infty} (\beta/\alpha)^k I_k(\alpha\beta)$$

and $I_k()$ is the modified Bessel function of first kind with order k.

$F_t(n)$ converges as t increases to the steady-state behavior. Therefore, the transient period of a M/M/1 queue can be estimated by setting n to different values and changing t (the run length) until $F_t(n)$ is above 0.9.

APPENDIX D

M/M/1 CONFIDENCE LEVEL

According to MacGougall[15], the queueing time of a M/M/1 queue tends to cluster into sequences of very long and very short time. Since the service time of M/M/1 queue is exponentially distributed, the majority of service times are short, less than the mean, but a few are very much longer. When one of these long service times occurs, customers back up in the server and incur long delays. Eventually a long inter-arrival time occurs, the server catches up and delays diminish. As a result, delays of a M/M/1 queue tend to be positively correlated; a long delay is more likely to be followed by another long delay rather than by a short one, and conversely. This correlation effect among the samples has created problems in estimating the confidence interval. If confidence interval is required to estimate based on N samples. When correlated, these N samples will provide less information than N independent samples. How much less depends on the degree of correlation, which is only available for M/M/1 queue. The degree of correlation for M/M/1 queue (ξ) is

$$\xi = (1+r)/(1-r) + (2r(3-r))/((2-r)(1-r))$$

where r is the corresponding traffic intensity factor. Let us assume that the sample size n is large enough, then the confidence interval half bandwidth (H) is

$$H = Z(\alpha/2)[\sigma^2/n]^{1/2}$$

where $Z(\alpha/2)$ is the upper $\alpha/2$ quantile of the normal distribution and $1 - \alpha$ is called the confidence level or confidence coefficient, typical values of which are 0.90 or 0.95. σ^2 is the variance of the queueing time distribution, which is equal to

$$\sigma^2 = r(2-r)[Ts/(1-r)]^2,$$ Ts is the mean service time.

With the above formula, knowing the sample size n will tell us the confidence interval of the sampling experiment.

APPENDIX E

NOTATIONS

S_N	the state of a switching network composed of N switches.
s_i	the state of switch i in the network.
q_{ij}	the queueing process governing output queue j of switch i.
p(i)	the number of input/output ports possessed by switch i.
λ_{ij}	the mean arrival rate to output queue j of switch i.
$1/\mu_{ij}$	the mean service time of the trunk server associated with output queue j of switch i.
ρ_{ij}	the traffic intensity factor associated with output queue j of switch i.
π_k	the path k in a network, it is conceptualized as a sequence of input/output queues.
λ_k	the mean external arrival rate to switch k;

P_{kr} the probability of switch k calling switch r;

λ_{ij}^{kr} the mean arrival rate to the output queue j of switch i due to the search message originated by switch k toward switch r;

$Path_{ki}$ the set of paths connecting switch k to switch i.

$Path_{kr}^{[ij]}$ the set of paths connecting switch k to switch r which does not contain the output queue j of switch i.

$Path_{kr}^{\{ij\}}$ the set of paths connecting switch k to switch r which contains the output queue j of switch i.

$Path_{kr}^{[i]}$ the set of paths connecting switch k to switch r without switch i as an intermediate switch;

$Path_{kr}^{i}$ the set of paths connecting switch k to switch r with switch i as an intermediate switch;

$Path_{kr}^{im}$ the subset of $Path_{kr}^{i}$, and each path of the subset terminates at port m of switch r;

$Path_{kr}^{[i]m}$ the subset of $Path_{kr}^{[i]}$, and each path of the subset terminates at port m of switch r;

$Prob_{kr}^{ic}$ the probability that path c is the shortest one out of the set $Path_{kr}^{i}$;

Hop_i the number of hops in path i of the network.

$Prob_{kr}^{[i]c}$ the probability of path c is the shortest one out of the set $Path_{kr}^{[i]}$;

PFC_{ij}^{kr} the probability that a search message sent from switch k to switch r in the output queue j of switch i will cause the arrival of the corresponding forward message.

$PFC_{ij}^{\#}$ the probability that a search message in the output queue j of switch i will cause the arrival of the corresponding forward message. The search message can not be generated by switch i.

PFC_{ij}^{*} the probability that a search message generated by switch i in its output queue j will cause the arrival of the corresponding forward message.

PBC_{ij}^{kr} the probability of putting a pseudo search message into the output queue j of switch i. This event occurs when the first search message, sent by switch k toward switch r, enters into switch i through port j.

n the number of forward messages.

m the number of backward messages.

$1/\mu_s$ the mean service time of a search message.

$1/\mu_f$ the mean service time of a search message + n forward messages.

$1/\mu_b$ the mean service time of a search message + m backward messages.

SER_{ij} the mean service time associated with output queue j of switch i, it is tuned with respect to the probability of the arrival of the search message alone, or the search message plus the call set-up messages, or the pseudo search message.